CMPE 230: Systems Programming, Spring 2022, Final

Problem 1 (16 pts)
Give the Linux commands that will do the following:

(a) List symbols from the static library libgeom.a .
(b) Compile the C file func.c in order to produce a positional independent object file that can be added to a shared library.
(c) Add the current directory to the path environment variable.
(d) Change the current directory two levels up .
(e) Copy all the files in the current directory starting with "cmpe230" and ending with ".txt" to a sub-directory named "MYDIR".
(f) Print the contents of a directory in long format.
(g) Display the contents of a file named "file.txt" page by page.
(h) Switch current user to john.

Problem 2 (20 pts)
Consider the following program prog.c. It is compiled with the command:
      g++ prog.c   (i.e. for 64-bit system)
What is the output of the program ?

```
#include <iostream>
using namespace std;

template <class T>
T func1(T x, T y) {
  return(x+y) ;
}

class A {
  private:
    int x ;

  public:
    A(int i) {
      x = i ;
      cout << x << endl ;
    }
    ~A() {
      x = x + x ;
      cout << x << endl ;
    }
    void g() {
      x = func1<int>(x,1) ;
    }
    int getx() {
      return(x) ;
    }
};
```

```
void func2(A * & pobj ) {
  A  fobj(1) ;

  pobj = new A(2) ;
  pobj = new A(3) ;
  pobj->g() ;

  fobj = *pobj ;
}

int main() {
  A   obj(4)  ;
  A  *pobj  ;

  cout << obj.getx() << endl ;
  func2(pobj) ;
  obj.g() ;
  cout << sizeof(pobj) << endl ;
  cout << obj.getx() << endl ;

  delete(pobj) ;
  return 0;
}
```

Write your answer Here:

Problem 3 (20 pts)

Consider the following C++ program:

```cpp
#include <iostream>
using namespace std;

class A  {
  private:
    int ax ;

  protected:
    int ay ;

  public:
    int az ;

    void setaz(int u) {
        az = ax+ay+az ;        /* (a) */
    }
};

class B : public  A  {
  private:
    int bx ;

  protected:
    int by ;

  public:
    int bz ;

    void setby(int u) {
      bx = u + 4*ax ;          /* (b) */
    }
};

class C : public  A  {
  protected:
    int cy ;

  public:
    int cz ;
};
```

```cpp
class D : public  C  {
  private:
    int dx ;

  protected:
    int dy ;
    void setdy(int u) {
      dy = u + 3*cy ;          /* (c) */
    }

  public:
    int dz ;

    void setdx1(int u) {
      dx = 2*u ;               /* (d) */
    }

    void setdx2(int u) {
      dx = u + 3*bx ;          /* (e) */
    }
};


int main() {

  D  myDobj ;
  C  myCobj ;

  myDobj.setdx1(4)             ; /* (f) */
  myDobj.setdx2(4)             ; /* (g) */
  myDobj.setdy(4)              ; /* (h) */
  cout << myDobj.dz << endl    ; /* (i) */
  myDobj.dy = 20               ; /* (j) */

  return(0)                    ;

}
```

Write OK if the statements (a-j) will compile successfully or write ERROR if the compiler will report an error.

(a)	(b)	(c)	(d)	(e)	(f)	(g)	(h)	(i)	(j)

Problem 4 (24 pts)

Write a QT program that will display and do the following:
- A line-edit widget for inputting a word,
- An LCD display object that shows the number of characters in the word that is entered in the line-edit object,
- A capitalize button which capitalizes the word in the line-edit object,
- A close button that quits the program.

Note that a QT string object can be capitalized using toUpper method.

Problem 5 (20 pts)

The following assembly code was produced by the GNU C compiler by compiling a C program with the `gcc -s -m32` command . The C program only had a main function with some variable definitions and statements. Only a part of the generated assembly code is shown below:

Generated GNU assembly code	C Code (write your answer here)
| ```
_main:
LFB1:
    pushl  %ebp
LCFI0:
    movl   %esp, %ebp
LCFI1:
    subl   $16, %esp
    call   ___x86.get_pc_thunk.ax
L1$pb:
    movl   $3, -4(%ebp)
    movl   $4, -8(%ebp)
    movl   $5, -12(%ebp)
    movl   -4(%ebp), %edx
    movl   -8(%ebp), %eax
    addl   %eax, %edx
    movl   -12(%ebp), %eax
    addl   %edx, %eax
    movl   %eax, -4(%ebp)
    movl   -4(%ebp), %edx
    movl   -12(%ebp), %eax
    addl   %edx, %eax
    movl   %eax, -8(%ebp)
    movl   -4(%ebp), %edx
    movl   -8(%ebp), %eax
    addl   %edx, %eax
    movl   %eax, -12(%ebp)
    movl   $0, %eax
    leave
LCFI2:
    ret
``` | |

Give the full C code that corresponds to the above GNU assembly code. (in other words, disassemble the GNU assembly code given above).

Problem 1

a) `nm libgeom.a`
b) `gcc —fpic func.c`
c) `export PATH=./$PATH`
d) `cd ../..`
e) `cp cmpe230*.txt MYDIR/`
f) `ls —l`
g) `more file.txt`
h) `su john`

Problem 2

4
4
1
2
3
8
8
5
8
10

Problem 3

a) ok
b) error
c) ok
d) ok
e) error
f) ok
g) ok
h) error
i) ok
j) error

Problem 4

Code written by each student

Problem 5

```c
#include <stdio.h>

int main()
{
    int x, y, z ;

    x = 3 ;
    y = 4 ;
    z = 5 ;

    x = x + y + z ;
    y = x + z ;
    z = x + y ;

    return(0) ;
}
```